DETAILED ACTION

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claims 12, 18, 19, 21, 44, 45, 47-50 rejected under 35 U.S.C. 102(b) as being anticipated by US 6360332 to Weinberg et al.

3. Referring to claim 12, Weinberg discloses creating a first state; creating a second state; building a state machine from the first and second states, the state machine being capable of executing a plurality of functions, each of the plurality of functions being implemented in code common to multiple parameters and specific to each of the plurality of functions (From line 15 of column 2, "a software-implemented testing tool for generating, running, and viewing the results of tests for testing transactional servers." Figure 7, "parameters". Wherein the tool is implemented in software (code) that is used, commonly, to perform functionality involving parameters, the code being specific to the function.),

wherein the plurality of functions includes at least one selected from the group consisting of editing, storing, loading, and displaying (From line 41 of column 2, "One inventive feature of the testing tool involves displaying the test to the user as a hierarchical node structure, such as a tree, in which steps of the test are represented as corresponding nodes. Preferably, each node is displayed as an icon that indicates the

type of the step, together with a textual step name or annotation. To edit the test, the user can select a node from the node structure to view and edit the node's properties. In the preferred embodiment, for example, the user can modify the data input value associated with a data entry step, or modify the verification condition associated with a verification step. The user can also preferably edit the test by performing such actions as dragging-and-dropping and deleting nodes. An important benefit of this feature is that it allows the user to generate and edit tests without the need to know a scripting or other programming language.");

loading test parameters of the multiple parameters for a test through utilization of one of the plurality of functions of the state machine; performing the test with the current values of the test parameters through utilization of the state machine; determining whether the test is complete; varying a value of one of the test parameters within a boundary of the one of the test parameters through utilization of one of the plurality of functions of the state machine and testing with the current values of the test parameters through utilization of the state machine until the test is complete (Figure 7. Further, from line 56 of column 2, "Another inventive feature of the testing tool involves the provision of a data table for allowing the user to conveniently specify data sets for running multiple iterations of a test. Using this feature, the user can, for example, record a particular transaction (e.g., looking up and checking the departure time for a particular flight), and then specify additional data sets (e.g., additional flight numbers and expected departure times) to be used to test the transaction. The data table is preferably in the form of a standard-format spreadsheet in which each row corresponds

to a single iteration of the test and each column contains the data values to be used with a parameterized step of the test. The use of a spreadsheet for this purpose allows the test author to use standard spreadsheet functions and/or existing spreadsheet data to specify data sets (which may include both input values and expected results) for testing recorded transactions. During execution of the test, the testing tool automatically uses the data sets within the data table to run multiple iterations of the test."); and

logging at least one state change of the state machine that occurs during the test (From line 8 of column 3, "The testing tool preferably stores the results of the verification steps in a separate results spreadsheet.").

4. Referring to claim 18, Weinberg discloses a look up table for providing type and value information for each of the multiple parameters (From line 56 of column 2 (with emphasis), "Another inventive feature of the testing tool involves the provision of a data table for allowing the user to conveniently specify data sets for running multiple iterations of a test. Using this feature, the user can, for example, record a particular transaction **(e.g., looking up and checking the departure time for a particular flight), and then specify additional data sets (e.g., additional flight numbers and expected departure times)** to be used to test the transaction. The data table is preferably in the form of a standard-format spreadsheet in which each row corresponds to a single iteration of the test and each column contains the data values to be used with a parameterized step of the test. The use of a spreadsheet for this purpose allows the test author to use standard spreadsheet functions and/or existing spreadsheet data to specify data sets (which may include both **input values** and expected results) for

testing recorded transactions. During execution of the test, the testing tool automatically uses the data sets within the data table to run multiple iterations of the test.").

5. Referring to claim 19, Weinberg discloses the type and value information includes a range of values that are permitted for each of the multiple parameters (From line 56 of column 2 (with emphasis), "Another inventive feature of the testing tool involves the provision of a data table for allowing the user to conveniently specify data sets for running multiple iterations of a test. Using this feature, the user can, for example, record a particular transaction (e.g., looking up and checking the departure time for a particular flight), and then specify additional data sets (e.g., additional flight numbers and expected departure times) to be used to test the transaction. The data table is preferably in the form of a standard-format spreadsheet in which **each row corresponds to a single iteration of the test and each column contains the data values to be used with a parameterized step of the test**. The use of a spreadsheet for this purpose allows the test author to use standard spreadsheet functions and/or existing spreadsheet data to specify data sets (which may include both input values and expected results) for testing recorded transactions. During execution of the test, the testing tool automatically uses the data sets within the data table to run multiple iterations of the test." Further, from line 35 of column 14, "The testing tool provides the option of using parameters to drive and verify a business process. Parameters provide a mechanism for specifying different data values (both input values and expected responses) to be used with different iterations of a test. For example, in a test which accesses a web site to check prices of airline tickets, a data entry step for specifying a

flight number may be parameterized to permit entry of multiple flight numbers (one per iteration), and a verification step for checking corresponding prices may be parameterized to require different dollar amounts or ranges for different flights." Further, that values span a range is inherent. The degree and granularity of that range may not be.).

6. Referring to claim 21, Weinberg discloses at least one of the multiple parameters is "independent of type" (It must be recognized that everything has something as vague as a "type". Even things that defy typification may be deemed of the "type" "untypified". Therefore, to be "independent" of "type" conveys a degree of vagueness that renders the limitation largely broad and extremely open to interpretation. That having been said, Weinberg discloses, from line 16 of column 2, "The various inventive features of the testing tool may be used separately or in combination to test the functionality of a variety of different types of transactional servers. In a preferred embodiment, the testing tool is used to test web-based, SAP-based and mainframe-based servers." Further, in view of claim 48, it is understood that Weinberg does not disclose the parameters to be dependent upon a bus. Further, see rejections of claims 14, 15.).

7. Referring to claim 44, Weinberg discloses adding at least one new parameter to the multiple parameters (From line 23 of column 2, "In a preferred embodiment, the testing tool generates tests by recording interactions between a user and the transactional server as the user performs a transaction, such as a business process. For example, in a web-based implementation, the testing tool records interactions between a web browser and a web server, including link selections and form

submissions made by the user and pages returned by the server. During or following the recording session, the user can define verification steps to test for expected server responses. For example, the user can define verification steps to test for expected text messages, images, or numerical values within a web page or other screen returned by the transactional server.").

8. Referring to claim 45, Weinberg discloses performing one selected from adding, changing, and deleting at least one state from the state machine (From line 23 of column 2, "In a preferred embodiment, the testing tool generates tests by recording interactions between a user and the transactional server as the user performs a transaction, such as a business process. For example, in a web-based implementation, the testing tool records interactions between a web browser and a web server, including link selections and form submissions made by the user and pages returned by the server. During or following the recording session, the user can define verification steps to test for expected server responses. For example, the user can define verification steps to test for expected text messages, images, or numerical values within a web page or other screen returned by the transactional server.").

9. Referring to claim 47, Weinberg discloses building the state machine further comprises importing parameter information into the code common to the multiple parameters and specific to each of the plurality of functions (From line 56 of column 2, "Another inventive feature of the testing tool involves the provision of a data table for allowing the user to conveniently specify data sets for running multiple iterations of a test. Using this feature, the user can, for example, record a particular transaction (e.g.,

looking up and checking the departure time for a particular flight), and then specify additional data sets (e.g., additional flight numbers and expected departure times) to be used to test the transaction. The data table is preferably in the form of a standard-format spreadsheet in which each row corresponds to a single iteration of the test and each column contains the data values to be used with a parameterized step of the test. The use of a spreadsheet for this purpose allows the test author to use standard spreadsheet functions and/or existing spreadsheet data to specify data sets (which may include both input values and expected results) for testing recorded transactions. During execution of the test, the testing tool automatically uses the data sets within the data table to run multiple iterations of the test.").

10. Referring to claim 48, Weinberg discloses the state machine is independent of bus type (Weinberg does not disclose the state machine is dependent on any particular bus.).

11. Referring to claim 49, Weinberg discloses a second state machine interrelated with the state machine (A state machine, particularly of the type disclosed in Weinberg, is comprised of state machines. Further, see all the flow diagrams of Weinberg. Further, anything that may "interrelate" with Weinberg's testing tool may be considered "a second state machine".).

12. Referring to claim 50, Weinberg discloses pointers and status functions are utilized to build and maintain the state machine (From line 61 of column 23, "In step 808, the testing tool receives a pointer to the object that was created. The object contains, in addition to the information regarding the graphical representation of the

function, attributes that are particular to the function that was converted to a node." Further, see all flow diagrams of Weinberg, which represent "state machines" whose states must be functionally tracked.).

Claim Rejections - 35 USC § 103

13. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

14. **Claim 14 rejected under 35 U.S.C. 103(a) as being unpatentable over Weinberg as applied to claim 12 above, and further in view of US 20030093608 to Jaramillo et al.**

15. Referring to claim 14, although Weinberg does not specifically disclose the multiple parameters include a PCI cache line size, adjusting a PCI cache line size is known in the art. An example of this is shown by Jaramillo from paragraph 44, "This approach can be implemented by using other multiples or with a programmable multiple, or the standard PCI specification cache line size register can be adjusted such that the PCI to PCI bridge 350 actually prefetches multiple cache lines." A person of ordinary skill in the art at the time of the invention would have been motivated to use PCI cache line size as a parameter because, from paragraph 44 of Jaramillo, "It raises the overall system performance dramatically." Further, such a parameter could have been included for testing because Weinberg discloses that the invention is broadly applicable to the testing of server functionality, which may obviously comprise PCI

cache functionality, from line 40 of column 1, "Therefore, the transactional servers and the business applications that run on the transactional servers should be properly tested to ensure reliable operation."

16. **Claim 15 rejected under 35 U.S.C. 103(a) as being unpatentable over Weinberg as applied to claim 12 above, and further in view of US 6675244 to Elliot et al.**

17. Referring to claim 15, although Weinberg does not specifically disclose the multiple parameters include a Small Computer System Interface (SCSI) synchronous rate, adjusting the SCSI system rate is known in the art. An example of this is shown by Elliot from line 34 of column 7, "The ASRT state 608 is a transitory state in which the timer is loaded with a value suitable for a delay discussed in conjunction with the next state, a WAIT_ASRT state 610. The value loaded into the timer during the ASRT state 608 depends on whether the linear mode is enabled, what the determined SCSI synchronous rate is, and whether this particular clock pulse is being "stretched". These aspects are further discussed below in conjunction with FIGS. 8-12. To summarize, if the linear mode is enabled, the SCSI clock will be asserted for a number of repeater 40 clock cycles that most closely matches the incoming clock signal from the other side of the repeater 40, but with some degree of "snapping" when the rate is near a standard SCSI rate." A person of ordinary skill in the art at the time of the invention would have been motivated to include a SCSI synchronous rate because, as disclosed by Elliot, the rate affects system performance. Further, such a parameter could have been included for testing because Weinberg discloses that the invention is broadly applicable to the

testing of server functionality, from line 40 of column 1, "Therefore, the transactional servers and the business applications that run on the transactional servers should be properly tested to ensure reliable operation."

18. **Claim 16 rejected under 35 U.S.C. 103(a) as being unpatentable over Weinberg as applied to claim 12 above, and further in view of "block size" by Microsoft Computer Dictionary (MSCD).**

19. Referring to claim 16, although Weinberg does not specifically disclose the multiple parameters include block size, adjusting the block size is known in the art. An example of this is shown by MSCD, "The declared size of a block of data transferred internally within a computer, via FTP, or by modem. The size is usually chosen to make most efficient use of all the hardware devices involved." A person of ordinary skill in the art at the time of the invention would have been motivated to include a block size because, as disclosed by MSCD, the block size affects system performance. Further, such a parameter could have been included for testing because Weinberg discloses that the invention is broadly applicable to the testing of server functionality, from line 40 of column 1, "Therefore, the transactional servers and the business applications that run on the transactional servers should be properly tested to ensure reliable operation."

20. Claim 17 rejected under 35 U.S.C. 103(a) as being unpatentable over US Weinberg as applied to claim 12 above, and further in view of US 6507842 to Grey et al.

21. Referring to claim 17, Weinberg discloses a table of values for the multiple parameters (Figure 7. Further, from line 56 of column 2, "Another inventive feature of

the testing tool involves the provision of a data table for allowing the user to conveniently specify data sets for running multiple iterations of a test. Using this feature, the user can, for example, record a particular transaction (e.g., looking up and checking the departure time for a particular flight), and then specify additional data sets (e.g., additional flight numbers and expected departure times) to be used to test the transaction. The data table is preferably in the form of a standard-format spreadsheet in which each row corresponds to a single iteration of the test and each column contains the data values to be used with a parameterized step of the test. The use of a spreadsheet for this purpose allows the test author to use standard spreadsheet functions and/or existing spreadsheet data to specify data sets (which may include both input values and expected results) for testing recorded transactions. During execution of the test, the testing tool automatically uses the data sets within the data table to run multiple iterations of the test.").

Although Weinberg does not specifically disclose that those values may be drawn from a "look-up table of default values", such a table is known in the art. An example of this is shown in Grey, from line 3 of column 3, "After including the Property Loader step in a sequence at edit time, the user may configure the step to load the desired variable and/or property values from the database. In various embodiments, this configuration process may require specifying various types of information. For example, the user may specify a particular database from which to load the values. The user may also specify a mapping of properties/variables to database values. For example, specifying this mapping may comprise specifying a database table and a mapping of

properties/variables to columns in the table. For other types of databases, e.g., object-oriented databases, this mapping may be specified in any of various other ways. Exemplary user interface dialog boxes for specifying the mapping are discussed. The user may also configure the Property Loader step with filtering information specifying criteria which the database values must satisfy in order to be loaded. If the filtering criteria are not specified, default values for the variables/properties may be used instead." A person of ordinary skill in the art at the time of the invention could have been motivated to use such a default value database because it provides values for parameters for which criteria are not specified, or at least not fully specified, as is the case in Weinberg, as cited above.

22. **Claim 20 rejected under 35 U.S.C. 103(a) as being unpatentable over Weinberg as applied to claim 19 above, and further in view of US 6546507 to Coyle et al.**

23. Referring to claim 20, although Weinberg does not specifically disclose the type and value information includes an incremental step size for each of the multiple parameters, using an incremental step for testing values is known in the art. An example of this is shown by Coyle, from line 46 of column 39, "The method 2850 starts at block 2852, where it verifies that the system operates correctly at a given initial value. The method 2850 in block 2854 tests whether the system passes at that value. If it does not, block 2856 reports a system failure. If the system passes, block 2858 adjusts the initial value to a new value, e.g., a single step up in the value. Block 2862 tests the system at this new value, and, if it passes, block 2862 saves the new value to a variable

called HIGHGOOD. Then, method 2850 returns to block 2858 where the value can be again incremented in the same direction. If the test of block 2860 fails, the method 2850 proceeds to block 2864, where it resets the parameter to the initial value. Then, block 2866 tests whether the system passes at this value. If it does not, then block 2868 reports a system failure. If the system passes, block 2872 adjusts the parameter in the opposite direction to that of block 2858, e.g., a single step down. In other words, one of the blocks 2858 and 2872 increments the parameter value to test the operational limit in one direction, while the other decrements that value to test the operational limit in the other direction." A person of ordinary skill in the art at the time of the invention would have been motivated to incrementally step because as disclosed by Coyle, it permits operational envelope testing, and further allows a type of testing that further verifies system operation.

24. **Claim 46 rejected under 35 U.S.C. 103(a) as being unpatentable over Weinberg as applied to claim 12 above, further in view of Official Notice.**

25. Referring to claim 46, although Weinberg does not specifically disclose utilizing the state machine to perform system level validation of new silicon, testing "new silicon" is very well known in the art. Examiner takes official notice for testing wafers, processors, circuits, or any such other computer accessible silicon, new. A person having ordinary skill in the art at the time of the invention could have been motivated to test such "new silicon" because it is new, and as such, may require validation, and further, the method of Weinberg shows a generality of applicability in testing by recording user interactions for modification and subsequent playback and verification.

Particularly in design validation, for which "new silicon" is presumed to be a recently fabricated circuit, there is a requirement for parameter and value testing, and reiteration, of the type disclosed by Weinberg.

Response to Arguments

26. Applicant's arguments with respect to claims 12, 14-21, 44-50 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

27. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. See notice of references cited.

28. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Gabriel L. Chu whose telephone number is (571) 272-3656. The examiner can normally be reached on weekdays between 8:30 AM and 5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Scott Baderman can be reached on (571) 272-3644. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Gabriel L. Chu/
Primary Examiner
Art Unit 2114

gc